



National Aeronautics and Space Administration

Engineering Elegant Systems: Design at the System Level

05 October 2017

Michael D. Watson, Ph.D.

Consortium Team

UAH

George Washington University

Iowa State

Texas A&M

University of Colorado at Colorado Springs (UCCS)

Missouri University of S&T

University of Michigan

Doty Consulting Services

AFRL Wright Patterson



Space Launch System



◆ Understanding Systems Engineering

- Postulates
- Hypothesis
- Principles

◆ Systems Engineering Domain

- System Integration
 - System State Variables
 - Goal Function Tree
 - State Analysis Model
 - System Value Model
 - System Integrating Physics
 - System Autonomy
 - Multidisciplinary Design Optimization (MDO)
 - Engineering Statistics
 - Methods of System Integration
- Discipline Integration
 - Sociological Concepts in Systems Engineering
 - Information Flow
 - Systems Thinking (Cognitive Science)
 - Policy and Law
 - System Dynamics

◆ Summary



Understanding Systems Engineering

- ◆ **System Engineering of Complex Systems is not well understood**
- ◆ **System Engineering of Complex Systems is Challenging**
 - System Engineering can produce elegant solutions in some instances
 - System Engineering can produce embarrassing failures in some instances
 - Within NASA, System Engineering does is frequently unable to maintain complex system designs within budget, schedule, and performance constraints
- ◆ **“How do we Fix System Engineering?”**
 - Michael D. Griffin, 61st International Astronautical Congress, Prague, Czech Republic, September 27-October 1, 2010
 - Successful practice in System Engineering is frequently based on the ability of the lead system engineer, rather than on the approach of system engineering in general
 - The rules and properties that govern complex systems are not well defined in order to define system elegance
- ◆ **4 characteristics of system elegance proposed as:**
 - System Effectiveness
 - System Efficiency
 - System Robustness
 - Minimizing Unintended Consequences

◆ Research Process

- Multi-disciplinary research group that spans systems engineering areas
- Selected researchers who are product rather than process focused

◆ List of Consortium Members

- Michael D. Griffin, Ph.D.
- Air Force Research Laboratory – Wright Patterson, Multidisciplinary Science and Technology Center: Jose A. Camberos, Ph.D., Kirk L. Yerkes, Ph.D.
- George Washington University: Zoe Szajnfarber, Ph.D.
- Iowa State University: Christina L. Bloebaum, Ph.D., Michael C. Dorneich, Ph.D.
- Missouri University of Science & Technology: David Riggins, Ph.D.
- NASA Langley Research Center: Anna R. McGowan, Ph.D., Peter A. Parker, Ph.D.
- The University of Alabama in Huntsville: Phillip A. Farrington, Ph.D., Dawn R. Utley, Ph.D., Laird Burns, Ph.D., Paul Collopy, Ph.D., Bryan Mesmer, Ph.D., P. J. Benfield, Ph.D., Wes Colley, Ph.D.
- Doty Consulting: John Doty, Ph.D.
- The University of Michigan: Panos Y. Papalambros, Ph.D.
- Ames Research Center: Peter Berg
- Glenn Research Center: Karl Vaden

◆ Previous Consortium Members

- Massachusetts Institute of Technology: Maria C. Yang, Ph.D.
- The University of Texas, Arlington: Paul Compton, Ph.D.
- Texas A&M University: Richard Malak, Ph.D.
- Tri-Vector Corporation: Joey Shelton, Ph.D., Robert S. Ryan, Kenny Mitchell
- The University of Colorado – Colorado Springs: Stephen B. Johnson, Ph.D.
- The University of Dayton: John Doty, Ph.D.
- Stevens Institute of Technology – Dinesh Verma
- Spaceworks – John Olds (Cost Modeling Statistics)
- Alabama A&M – Emeka Dunu (Supply Chain Management)
- George Mason – John Gero (Agent Based Modeling)
- Oregon State – Irem Tumer (Electrical Power Grid Robustness)
- Arkansas – David Jensen (Failure Categorization)

~40 graduate students and 5 undergraduate students supported to date

Understanding Systems Engineering



- ◆ **Definition – System Engineering is the engineering discipline which integrates the system functions, system environment, and the engineering disciplines necessary to produce and/or operate an elegant system.**

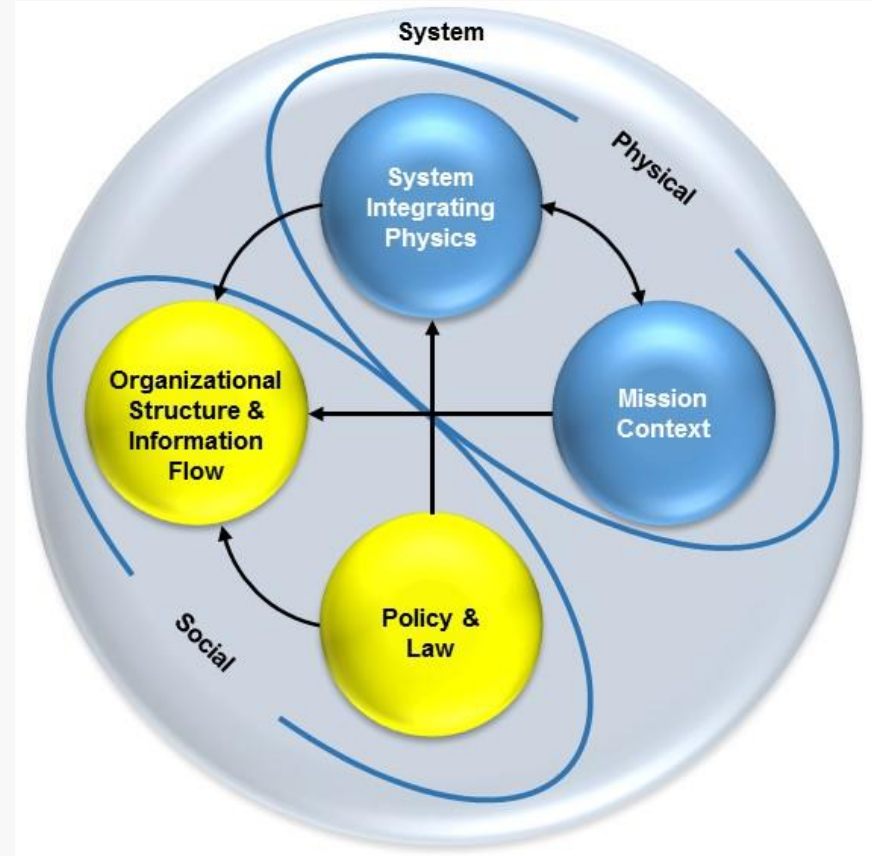
- **Elegant System** - A system that is robust in application, fully meeting specified and adumbrated intent, is well structured, and is graceful in operation.

- ◆ **Primary Focus**

- **System Design and Integration**
 - Identify system couplings and interactions
 - Identify system uncertainties and sensitivities
 - Identify emergent properties
 - Manage the effectiveness of the system
- **Engineering Discipline Integration**
 - Manage flow of information for system development and/or operations
 - Maintain system activities within budget and schedule

- ◆ **Supporting Activities**

- Process application and execution

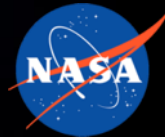


Systems Engineering Postulates



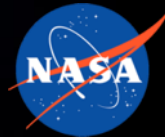
- ◆ **Postulate 1: Systems engineering is product specific.**
- ◆ **Postulate 2: The Systems Engineering domain consists of subsystems, their interactions among themselves, and their interactions with the system environment**
- ◆ **Postulate 3: The function of Systems Engineering is to integrate engineering disciplines in an elegant manner**
- ◆ **Postulate 4: Systems engineering influences and is influenced by organizational structure and culture**
- ◆ **Postulate 5: Systems engineering influences and is influenced by budget, schedule, policy, and law**
- ◆ **Postulate 6: Systems engineering spans the entire system life-cycle**
- ◆ **Postulate 7: Understanding of the system evolves as the system development or operation progresses**

Systems Engineering Principles



- ◆ **Principle 1: Systems engineering integrates the system and the disciplines considering the budget and schedule constraints**
- ◆ **Principle 2: Complex Systems build Complex Systems**
- ◆ **Principle 3: The focus of systems engineering during the development phase is a progressively deeper understanding of the interactions, sensitivities, and behaviors of the system**
 - Sub-Principle 3(a): Requirements reflect the understanding of the system
 - Sub-Principle 3(b): Requirements are specific, agreed to preferences by the developing organization
 - Sub-Principle 3(c): Requirements and design are progressively defined as the development progresses
 - Sub-Principle 3(d): Hierarchical structures are not sufficient to fully model system interactions and couplings
 - Sub-Principle 3(e): A Product Breakdown Structure (PBS) provides a structure to integrate cost and schedule with system functions
- ◆ **Principle 4: Systems engineering spans the entire system life-cycle**
 - Sub-Principle 4(a): Systems engineering obtains an understanding of the system
 - Sub-Principle 4(b): Systems engineering models the system
 - Sub-Principle 4(c): Systems engineering designs and analyzes the system
 - Sub-Principle 4(d): Systems engineering tests the system
 - Sub-Principle 4(e): Systems engineering has an essential role in the assembly and manufacturing of the system
 - Sub-Principle 4(f): Systems engineering has an essential role during operations and decommissioning

Systems Engineering Principles



◆ Principle 5: Systems engineering is based on a middle range set of theories

- Sub-Principle 5(a): Systems engineering has a mathematical basis
 - Systems Theory Basis
 - Decision & Value Theory Basis (Decision Theory and Value Modeling Theory)
 - Model Basis
 - State Basis (System State Variables)
 - Goal Basis (Value Modeling Theory)
 - Control Basis (Control Theory)
 - Knowledge Basis (Information Theory)
 - Predictive Basis (Statistics and Probability)
- Sub-Principle 5(b): Systems engineering has a physical/logical basis specific to the system
- Sub-Principle 5(c): Systems engineering has a sociological basis specific to the organization

◆ Principle 6: Systems engineering maps and manages the discipline interactions within the organization

◆ Principle 7: Decision quality depends on the system knowledge represented in the decision-making process

◆ Principle 8: Both Policy and Law must be properly understood to not overly constrain or under constrain the system implementation

◆ Principle 9: Systems engineering decisions are made under uncertainty accounting for risk

Systems Engineering Principles



- ◆ **Principle 10: Verification is a demonstrated understanding of all the system functions and interactions in the operational environment**
 - Ideally requirements are level and balanced in their representation of system functions and interactions
 - In practice requirements are not balanced in their representation of system functions and interactions
- ◆ **Principle 11: Validation is a demonstrated understanding of the system's value to the system stakeholders**
- ◆ **Principle 12: Systems engineering solutions are constrained based on the decision timeframe for the system need**

System Engineering Hypotheses



◆ Hypothesis 1: If a solution exists for a specific context, then there exists at least one ideal Systems Engineering solution for that specific context

- Hamilton's Principle shows this for a physical system

$$-\int_{t_1}^{t_2} (\delta T - \delta V + \delta W) dt = 0$$

- Kullback-Liebler Information shows this for ideal information representations of systems

$$-I(f, g) = \int f(x) \log(f(x)) dx - \int f(x) \log(g(x|\theta)) dx = 0$$

◆ Hypothesis 2: System complexity is greater than or equal to the ideal system complexity necessary to fulfill all system outputs

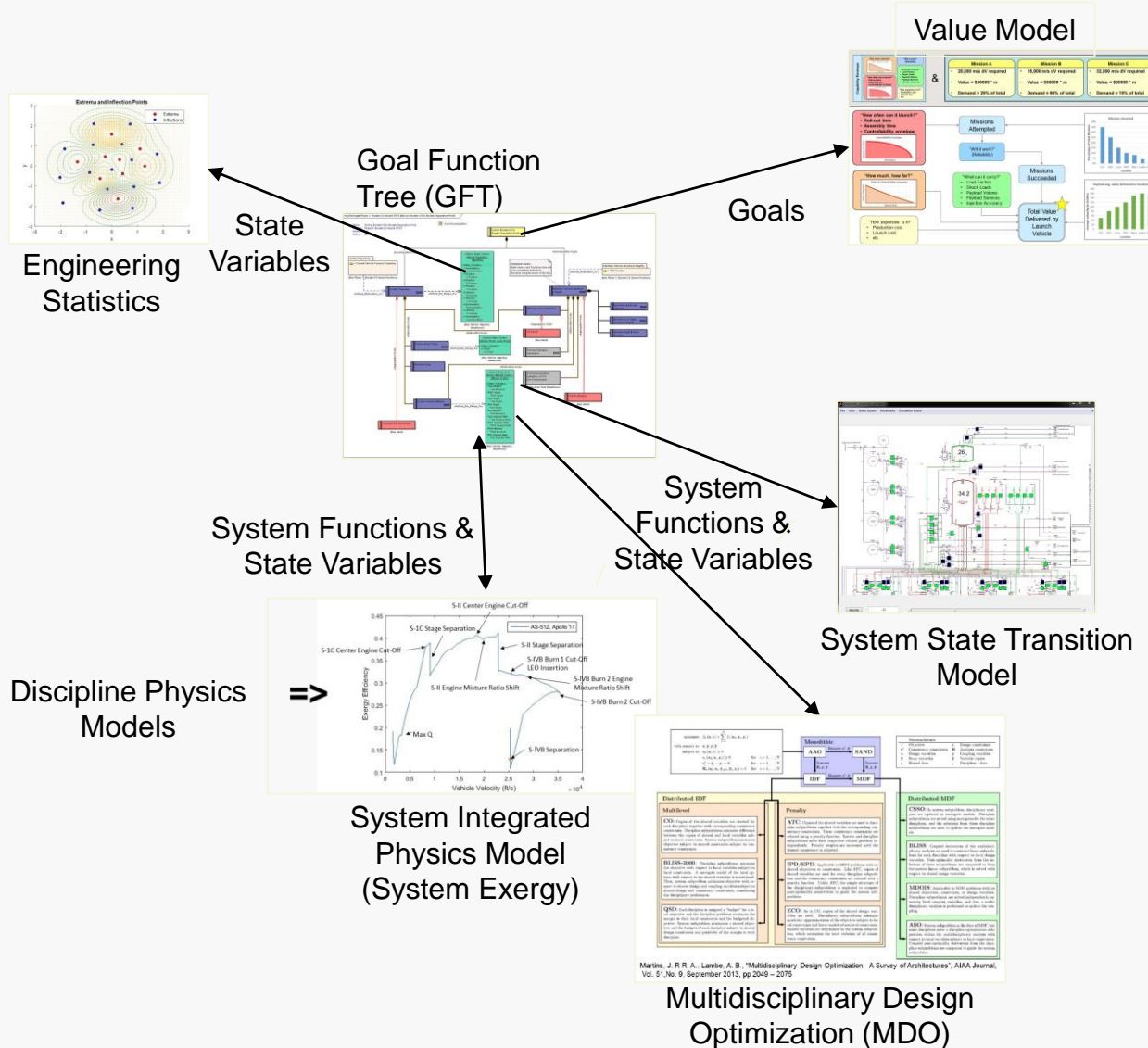
◆ Hypothesis 3: Key Stakeholders preferences can be accurately represented mathematically



Methods of System Design and Integration

**Goal: Techniques to Enable Integrated System
Design and Assessments by the Systems Engineer**

System Models Contain an Understanding of the System



- Allow systems engineers to:
 - Define system functions based on the system state variables
 - Understand stakeholders expectations on system value (i.e., capabilities)
 - Integrate discipline engineering models into a system level physics based model (e.g., system exergy)
 - Design and Analyze system responses and behaviors at the System level

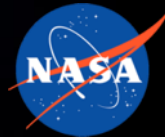
- MagicDraw Enterprise (SysML)
- Matlab
- Matlab StateFlow
- Microsoft Excell

System State Variables

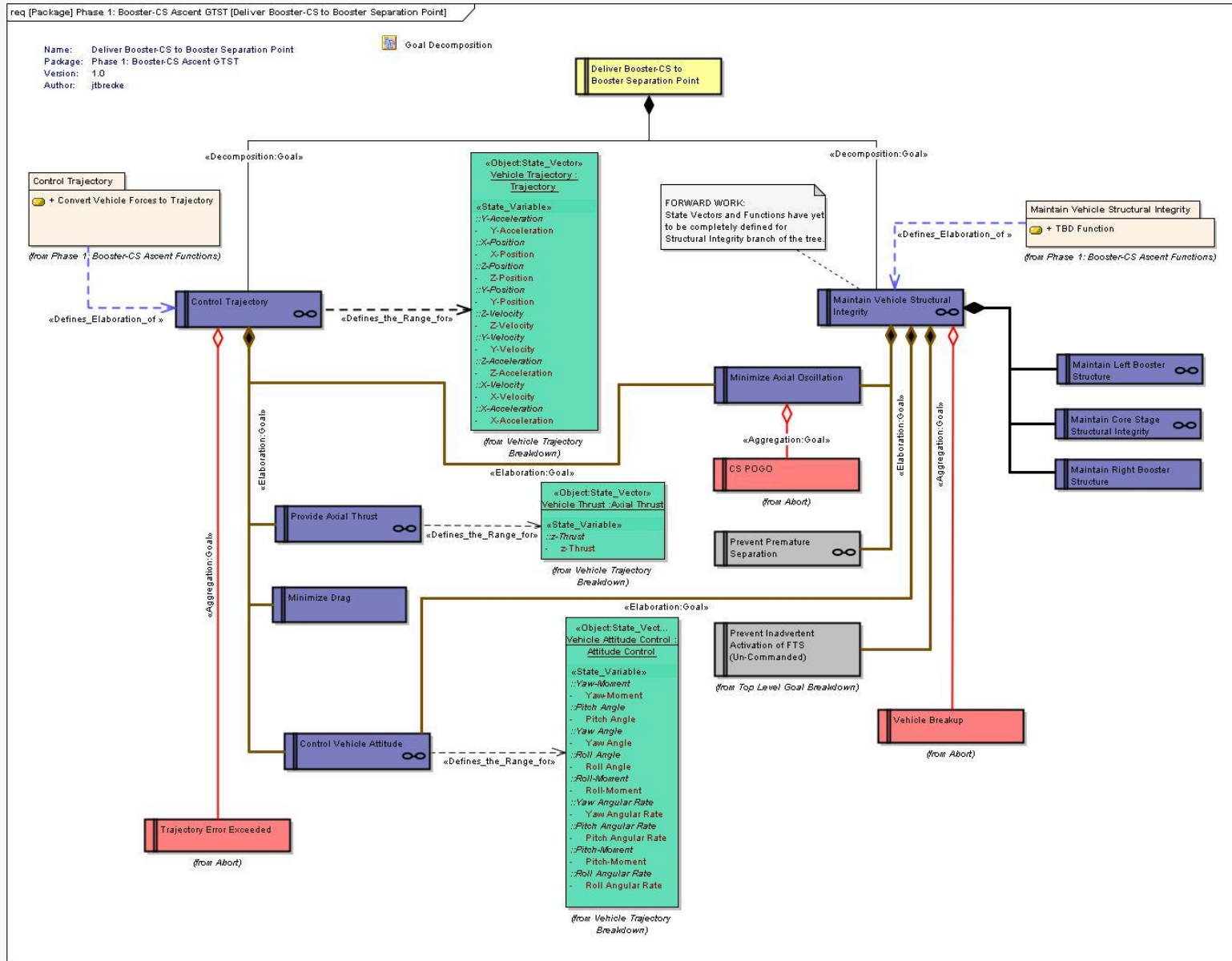
Goal: Utilize system state variables to understand the interactions of the system in relation to system goals and system execution

- ◆ **System Stage Models** represent the system as a whole in terms of the hardware and software states that the system transitions through during operation
- ◆ **Goal Function Tree (GFT) Model**
 - “Middle Out” model of the system based on the system State Variables
 - Shows relationship between system state functions (hardware and software) and system goals
 - Does not contain system physical or logical relationships and is not executable
- ◆ **System State Machine Model**
 - Models the integrated State Transitions of the system as a whole (i.e., hardware states and software states)
 - Confirms system functions as expected
 - Checks for system hazardous, system anomalies, inconsistent state progression, missing states, improper state paths (e.g., short circuits in hardware and/or software design)
 - Confirms that the system states progress as stated in the system design
 - Executable model of system

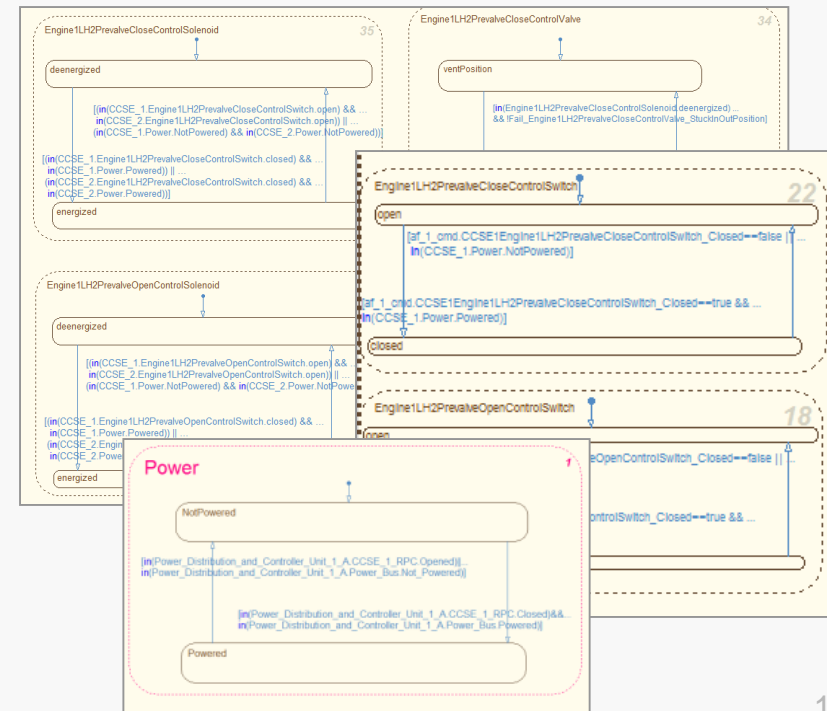
Booster – CS Ascent GFT



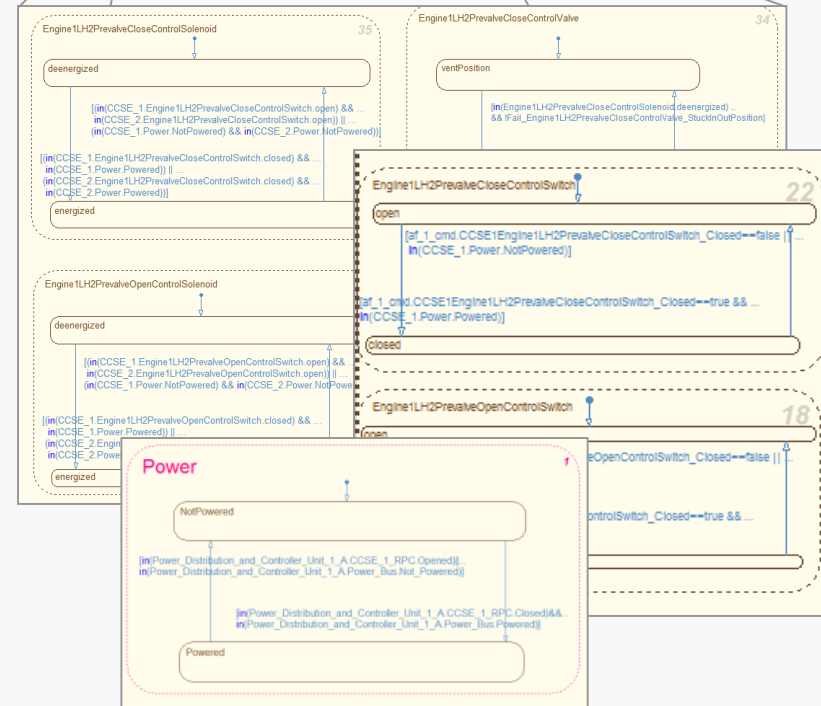
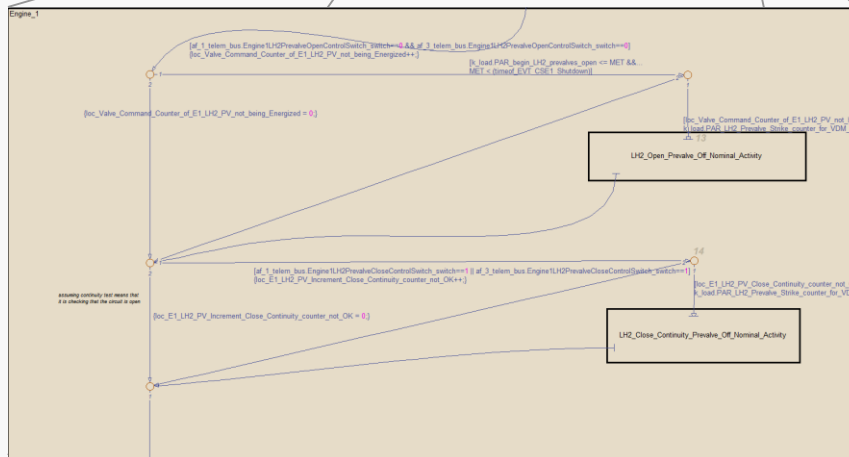
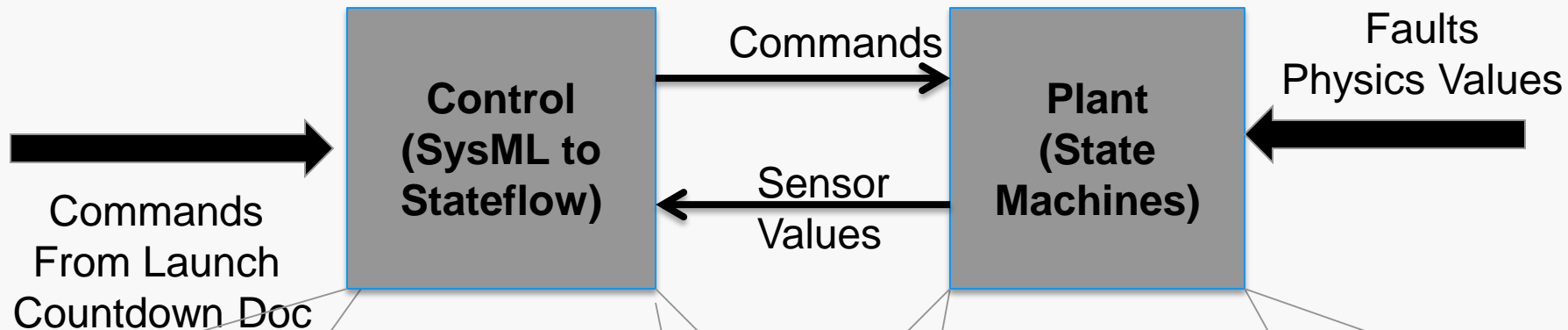
System Works



-
- The diagram illustrates the state machine logic for the LQ2_PV system. It features three states: **State 1** (initial), **State 2**, and **State 3**.
- State 1:**
 - Initial state.
 - Transition to State 2: Triggered by `[! 1_telem_bus.Engine.LQ2PrevalveOpenControlSwitch_switch==AA && ! 3_telem_bus.Engine.LQ2PrevalveOpenControlSwitch_switch==0]` and `[loc_Valve_Command_Counter_of_E1_LH2_PV_not_being_Energized++]`. Action: `[! load_PAR_begin_LH2_prevalves_open <= MET && MET < timed_EVT_CSBT_Shutdown]`.
 - Transition to State 3: Triggered by `[loc_Valve_Command_Counter_of_E1_LH2_PV_not_being_Energized = 5]`.
 - State 2:**
 - Transition to State 3: Triggered by `[! 1_telem_bus.Engine.LQ2PrevalveCloseControlSwitch_switch==1] & 3_telem_bus.Engine.LQ2PrevalveCloseControlSwitch_switch==1]` and `[loc_E1_LH2_PV_increment_Close_Continuity_counter_not_Ok++]`. Action: `[loc_E1_LH2_PV_increment_Close_Continuity_counter_not_Ok = 8]`.
 - State 3:**
 - Transition to State 2: Triggered by `[loc_E1_LH2_PV_Close_Continuity_counter_not_Ok < 8]` and `[! load_PAS_LH2_Prevails_State_counter_for_VDM_start]`. Action: `[! load_PAS_LH2_Prevails_State_counter_for_VDM_start]`.
- Activities shown in the diagram:
- LH2_Open_Prevail_Of_Nominal_Activity:** Associated with the transition from State 1 to State 2.
 - LH2_Class_Continuity_Prevail_Of_Nominal_Activity:** Associated with the transition from State 2 to State 3.
- Annotation: `continuity counter means that it is checking that the control is open` (pointing to the continuity counter logic).



State Analysis Model for SLS M&FM



- 14% of R12 modeled
- Over 7,200 Transitions in the Vehicle and Software
- Over 3,500 States in the Vehicle

System Value

Goal: Utilize system state variables to understand the interactions of the system in relation to system goals and system execution

System Value Model



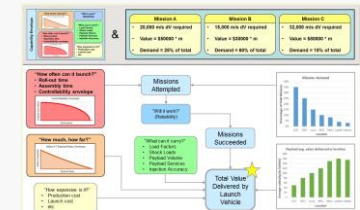
◆ A System Value Model is a mathematical representation of Stakeholders Preferences (Expectations) for the system

- The basic structure is straight forward
- The sociology/psychology of representing the Preferences can be a challenge

	Status	Gradient	Value
Efficiency	90%	150,000	135,000
Weight	700	-130	-91,000
Reliability	1500	2	3,450
Maintainability	7.8	-340	-2,652
Maintenance Cost	500	-1	-250
Support Equipment	12	-15	-180
Manufacturing Cost	700	-1	-700
Design Value			\$ 43,668

◆ The System Value Model is the Basis of System Validation!!!

- The Requirements and Design Models form the basis of System Verification
- The System Value Model forms the basis of System Validation



◆ Constructing an SLS Value Model to compare to System Validation results

- Can expand to Integrated Stack with input from MPCV and GSDO

◆ System Value model also provides basis for a measure of System Robustness

- How many mission types are supported by the system?

$$\pi = \int_{\text{aircraft}} x_1, x_2, \dots, x_n$$

$$v_e = \sum_{j=1}^m \left(\sum_{i=1}^n \frac{\partial \pi}{\partial x_i} \cdot \frac{\partial x_i}{\partial y_j} y_j \right)$$

$$v_t = \sum_{k=1}^p \left(\sum_{j=1}^m \frac{\partial v_e}{\partial y_j} \cdot \frac{\partial y_j}{\partial z_k} z_k \right)$$

System Physics and System Integrating Physics

Goal: Utilize the key system physics to produce an elegant system design

System Integrating Physics



- ◆ **Consortium is researching the significance of identifying and using the System Integrating Physics for Systems Engineering**
 - First Postulate: Systems Engineering is Product Specific.
 - States that the Systems are different, and therefore, the Integrating Physics for the various Systems is different
- ◆ **Launch Vehicles**
 - Thermodynamic System
- ◆ **Spacecraft**
 - Robotic
 - Integrated through the bus which is a thermodynamic system
 - Each Instrument may have a different integrating physics but integrates with the bus thermodynamically
 - Crew Modules
 - Integrated by the habitable volume (i.e., ECLSS)
 - A thermodynamic system
 - Entry, Descent, and Landing (EDL)
 - Integrated by thermodynamics as spacecraft energy is reduced in EDL
- ◆ **Other Thermodynamic Systems**
 - Fluid Systems
 - Electrical Systems
 - Power Plants
 - Automobiles
 - Aircraft
 - Ships
- ◆ **Not all systems are integrated by their Thermodynamics**
 - Optical Systems
 - Logical Systems
 - Data Systems
 - Communication Systems
 - Biological Systems
- ◆ **System Integrating Physics provides the engineering basis for the System Model**

Launch Vehicle and Crew Module System Exergy Balance



Launch Vehicle Exergy Balance

$$\Delta m_{prop} \sum_{stage} \left(h_{prop} + \frac{V_e^2}{2} \right) - X_{des}$$

$$= \left(M_{vehicle,final} \frac{V_{vehicle,final}^2}{2} - M_{vehicle,initial} \frac{V_{vehicle,initial}^2}{2} \right)$$

$$+ \left(\frac{GM_E M_{vehicle,initial}}{r_{altitude,initial}} - \frac{GM_E M_{vehicle,final}}{r_{altitude,final}} \right)$$

$$\eta_{ex} = 1 - \frac{X_{des}}{X_{expended}}$$

Crew Module Exergy Balance

$$\Delta X_{ECLSS} = \Delta X_{ACS} + \Delta X_{AR} + \Delta X_{THC} + \Delta X_{WRM} + \Delta X_{WM}$$

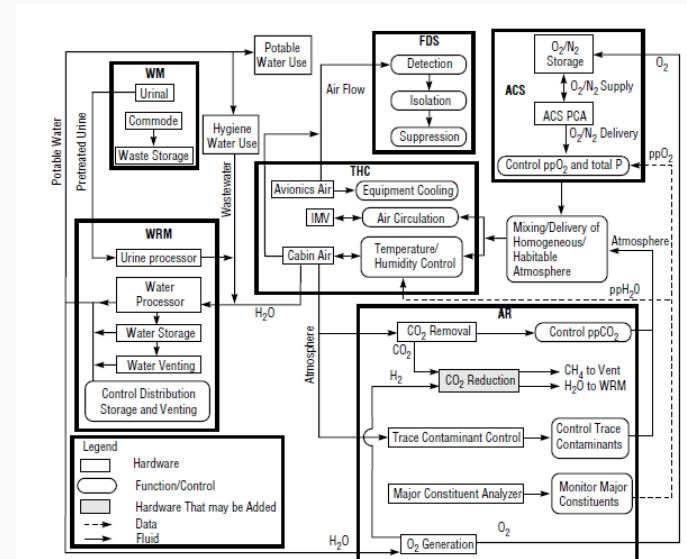
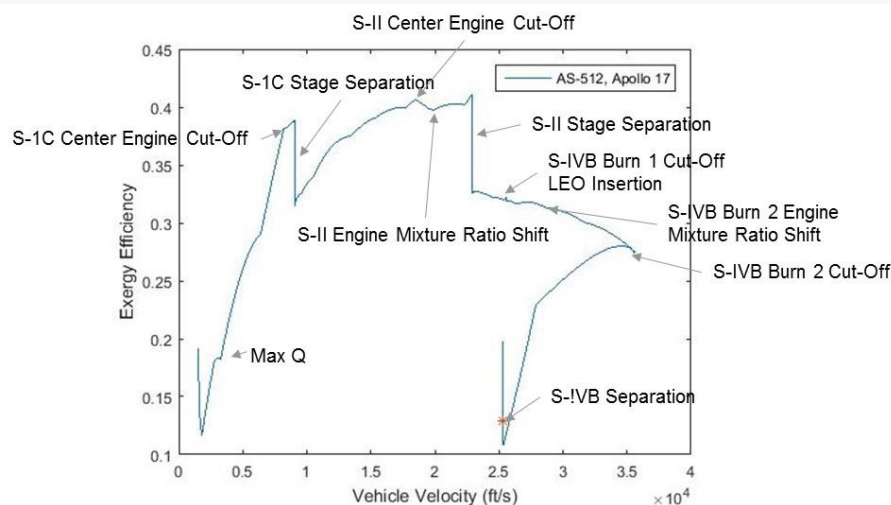
$$\sum_{process} m_{fluid} \left((h_{final} - h_{cabin}) - T_{cabin}(s_{final} - s_{cabin}) + \left(\frac{V_{final}^2}{2} \right) \right)$$

$$- \sum_{process} m_{fluid} \left((h_{initial} - h_{cabin}) - T_{cabin}(s_{initial} - s_{cabin}) + \left(\frac{V_{initial}^2}{2} \right) \right)$$

$$= \sum \left(1 - \frac{T_{cabin}}{T_{crew}} \right) Q_{crew} - \sum \left(\frac{T_{cabin} - T_{coolant}}{T_{coolant}} \right) Q_{TMS} + \sum W_{EPS}$$

$$- P_{cabin}(Vol_{final} - Vol_{initial})$$

$$+ m_{in} \left[\sum (h_{in} - h_{cabin}) - T_{cabin}(s_{in} - s_{cabin}) + \left(\frac{V_{in}^2}{2} \right) \right]$$



Spacecraft Exergy Balance and Optical Transfer Function



Spacecraft Exergy Balance

$$\begin{aligned} & \Delta m_{\text{propellant,engine}} \left(h_{\text{prop,engine}} + \frac{V_{e,\text{engine}}^2}{2} \right) \\ & + \Delta m_{\text{propellant,thruster}} \left(h_{\text{prop,thruster}} + \frac{V_{e,\text{thruster}}^2}{2} \right) \\ & + \sum_t (\sigma A e (T_{\text{radiator}}^4 - T_{\text{space}}^4) + V_{\text{bus}} I_{\text{bus}} \cos(\theta)) \Delta t - X_{\text{des}} \\ & = \left(M_{\text{vehicle,final}} \left(\frac{I_{c,\text{final}} \omega_{\text{vehicle,final}}^2}{2} + \frac{V_{\text{vehicle,final}}^2}{2} \right) \right) \end{aligned}$$

Optical Transfer Function

$$\begin{aligned} & \iint_{-\infty}^{\infty} \psi_{\text{obj}} S_f dx dy \\ & = \iint_{-\infty}^{\infty} \psi_{\text{obj}}(x_0 + \epsilon x, y_0 + \epsilon y) e^{j \left(\frac{k_0}{2f_1} \right) (x^2 + y^2)} \text{circ}(x + \Delta x + \delta x, y + \Delta y + \delta y) dx dy \end{aligned}$$

Where

$$\epsilon x = 1.22 \lambda_0 \left(\frac{f_1}{d_0 + \epsilon z + \omega_y \Delta t} \right) + v_x \Delta t + \omega_y \Delta t$$

$$\epsilon y = 1.22 \lambda_0 \left(\frac{f_1}{d_0 + \epsilon z + \omega_x \Delta t} \right) + v_y \Delta t + \omega_x \Delta t$$

$$f_1 = -\frac{R}{2} = -\frac{\sqrt{(x + \Delta x + \delta x)^2 + (y + \Delta y + \delta y)^2 + (z + \Delta z + \delta z)^2}}{2}$$

$$\Delta x = \alpha x \Delta T$$

$$\Delta y = \alpha y \Delta T$$

$C^2 > 4Mk$ Over Damped

$$\delta x = c_1 e^{-\left(\frac{C}{2M} - \frac{1}{2M} \sqrt{C^2 - 4Mk}\right)t} + c_2 e^{-\left(\frac{C}{2M} + \frac{1}{2M} \sqrt{C^2 - 4Mk}\right)t}$$

$C^2 = 4Mk$ Critically Damped

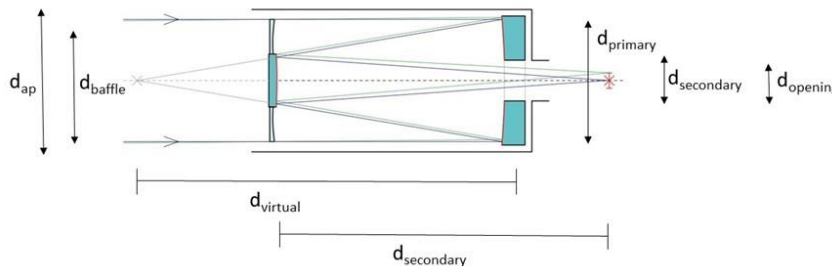
$$\delta x = (c_1 + c_2) e^{-\left(\frac{C}{2M}\right)t}$$

$C^2 < 4Mk$ Under Damped

$$\delta x = c_3 e^{-\left(\frac{C}{2M}\right)t} \cos\left(\sqrt{4Mk - C^2}t - \varphi\right)$$

$$\tan(\varphi) = \frac{x'(0)}{x(0) \sqrt{\frac{k}{M}}}$$

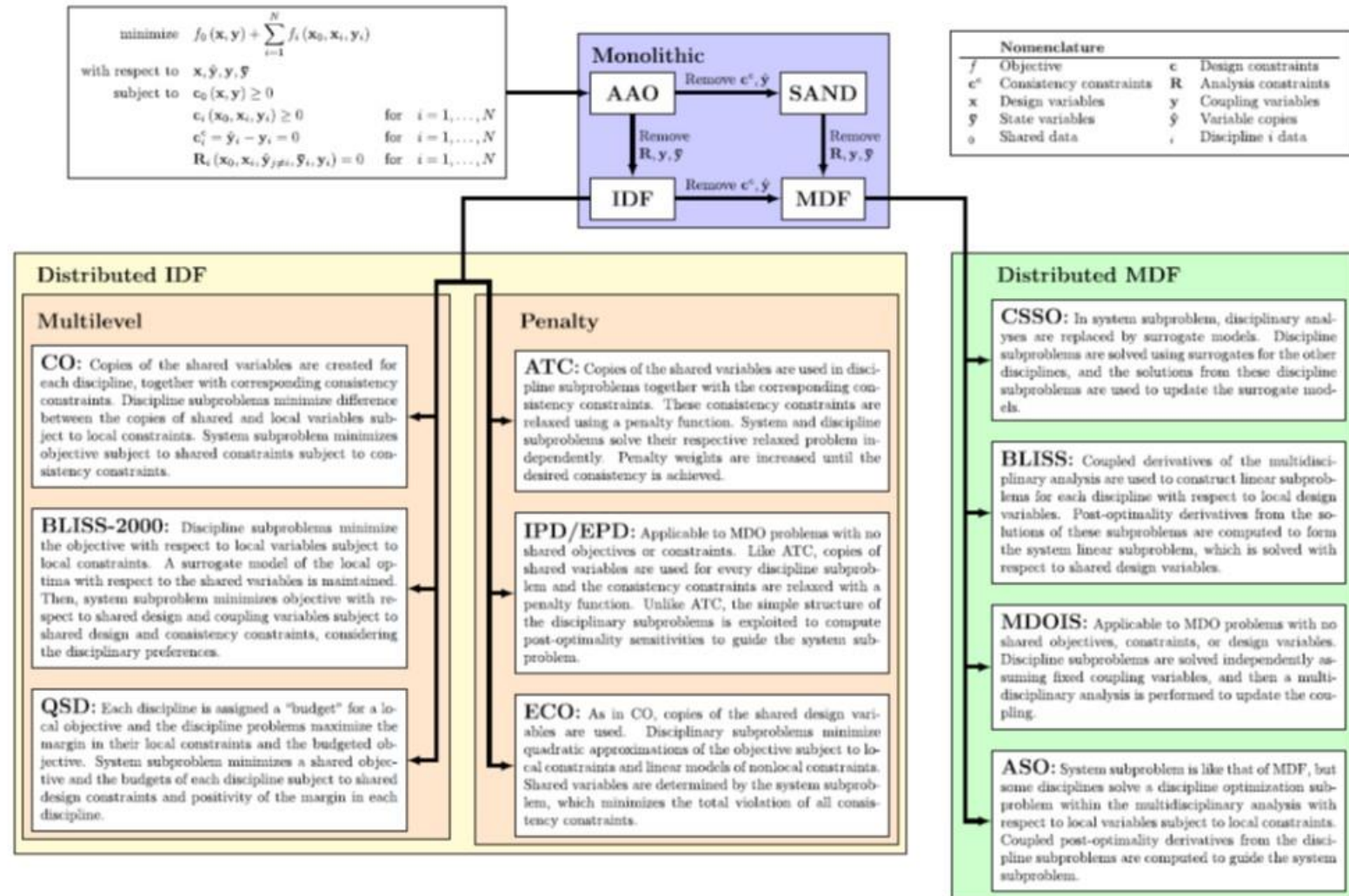
$$c_3^2 = \sqrt{x(0)^2 + \frac{M}{k} x'(0)^2}$$



System Design and Optimization

**Goal: Apply system design and optimization tools
to understand and engineer system interactions**

Multidisciplinary Design Optimization



Engineering Statistics

Goal: Utilize statistical methods to understand system uncertainties and sensitivities

Systems Engineering makes use of Frequentist Approaches, Bayesian Approaches, Information Theoretic Approaches as appropriate

Optimal Sensor Information Configuration



- ◆ Applying Akaike Information Criteria (AIC) corrected (AICc) to assess sensor coverage for a system

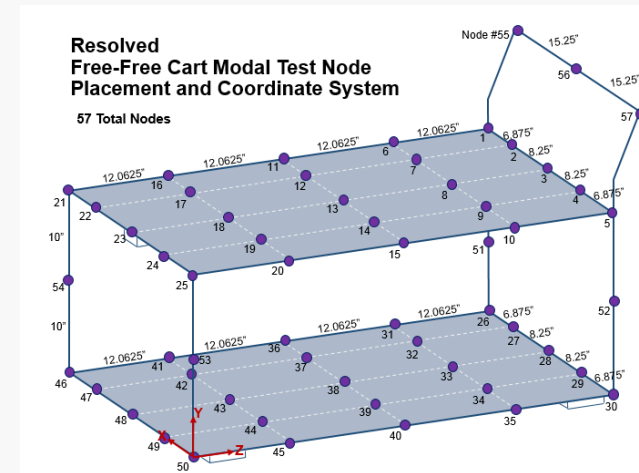
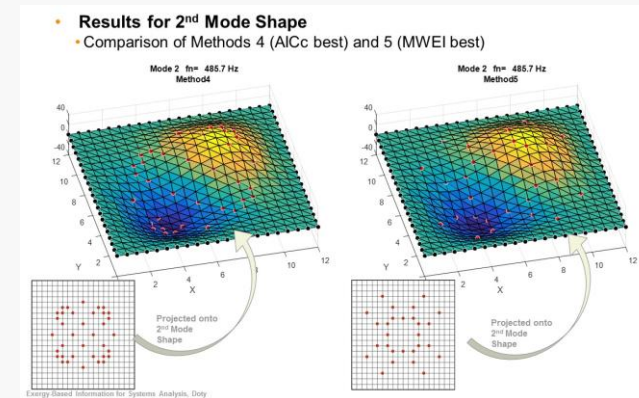
$$AICc(F) = -2 \left(I^{KL}(F|G) \right) + 2K + \frac{2K(K+1)}{n - K - 1}$$

- ◆ Two Views of Information Content

- AIC Information
 - Information is viewed as the number of meaningful parameters
 - Parameters with sufficient measurements to be reasonable estimates
- Fisher Information Matrix
 - Defines information as the matrix of partial second derivatives
 - Information is the amount of parameters with non zero values (so provides an indication of structure)
 - This value converges to a maximum as the number of parameters goes to infinity
 - Does not contain an optimum, always increases with added parameters

- ◆ AIC/AICc has an adjustment factor to penalize sensor arrangements where:
number of sensors < 3x(number of measurements)

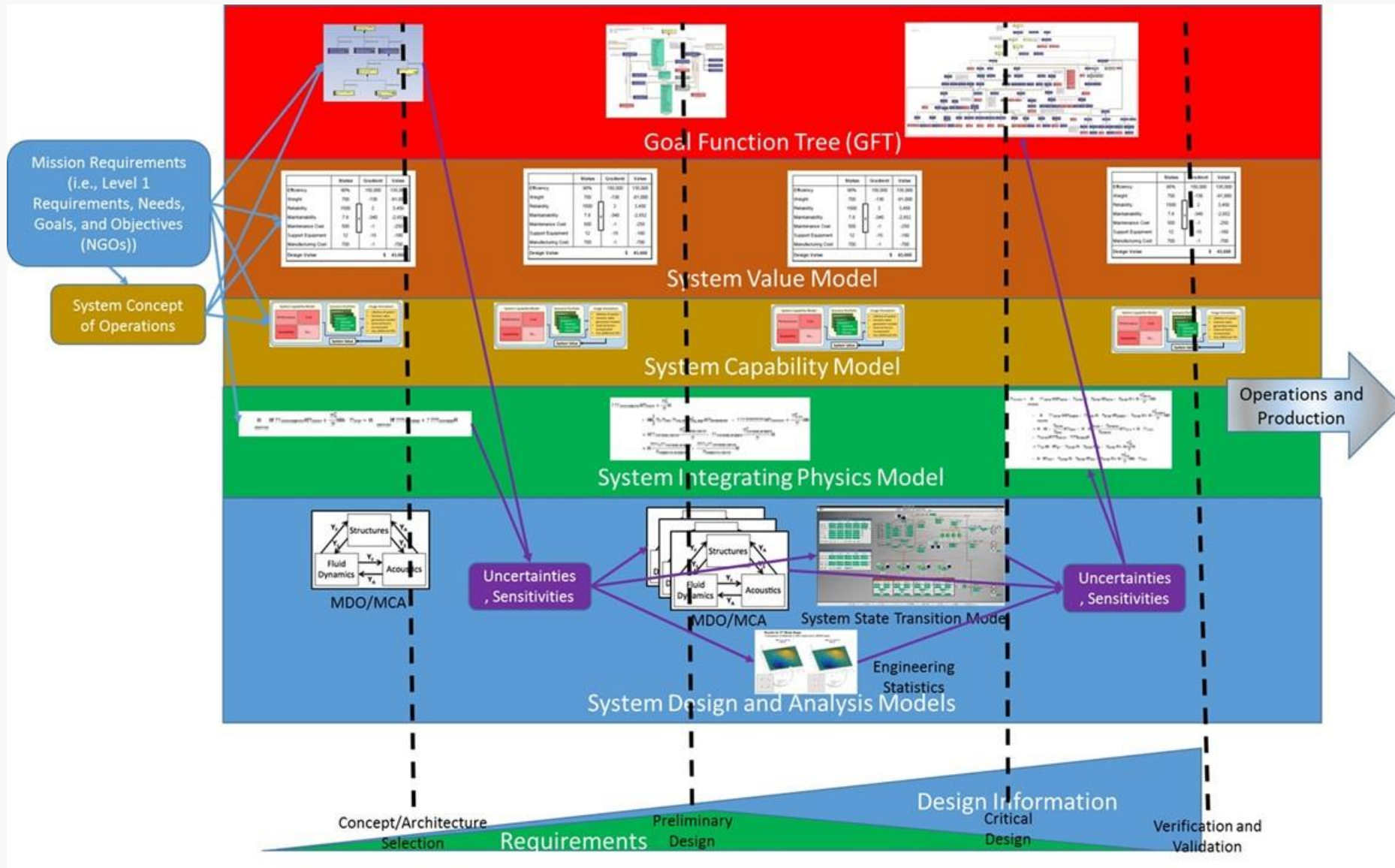
- ◆ Provides an optimization tool for use with System Models



Methods of System Integration

Goal: System Design and Analysis

System Design and Integration



Methods of Engineering Discipline Integration

**Goal: Understand How Organizational Structures
influence Design and Operations Success of
Complex Systems**

Sociological Concepts in Systems Engineering



- ◆ **Specification of Ignorance is important in the advancement of the understanding of the system**
- ◆ **Consistent use of Terminology is important for Communication within the Organization**
- ◆ **Opportunity Structures**
 - Provide opportunity to mature ideas
 - Task teams, working groups, communities of practice, etc.
- ◆ **Socially Expected Durations will exist about the project**
- ◆ **Both Manifest and Latent Social Functions exist in the organization**
- ◆ **Social Role Sets**
 - Individuals have a set of roles for their position
- ◆ **Cultural Subsets will form**
 - i.e., disciplines can be a subset within the organization
 - Insider and Outsider attitudes can form
 - Be Aware of the Self-Fulfilling Prophecy, Social Polarization
- ◆ **Reconsiderations Process (i.e., Reclama Process)**
 - Provides ability to manage social ambivalence
 - Must be able to recognize social beliefs that may be contributing to the disagreement
 - Helps to avoid putting people in to social dysfunction or complete social anomie
 - Conformity
 - Innovation
 - Ritualism
 - Retreatism
 - Rebellion

Unintended Consequences

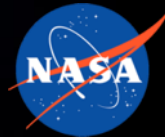


- ◆ **Unintended Consequences are the result of human mistakes.**
 - Physics do not fail, we do not recognize the consequences.

- ◆ **Based on sociology, followed the work of Robert K. Merton in classifying unintended consequences.**
 - “The Unanticipated Consequences of Social Action”, 1936

- ◆ **Classification**
 - Ignorance (limited knowledge of the problem)
 - Historical Precedent (confirmation bias)
 - Error (mistakes in calculations, working from habit)
 - Short Sightedness (imperious immediacy of interest, focusing on near term and ignoring long term consequences)
 - Cultural Values (cultural bias in what can and cannot happen)
 - Self Defeating Prophecy (by stating the hypothesis you induce a set of conditions that prevent the hypothesis outcome)

Information Flow



◆ Information Flow through a program/project/activity is defined by Information Theory

- Organizational communication paths
- Board Structure

◆ Decision Making follows the First Postulate

- Decision Process is specific to the decision being made
- Tracked 3 SLS CRs, with 3 separate task team processes, all had equally rated effectiveness

SLS SE&I MANAGEMENT STRUCTURE										
June 9, 2014 version										
SLS PROGRAM OFFICE ORGANIZATION	CHIEF ENGINEERS OFFICE ORGANIZATION	() = DMR () = ORGANIZATIONS MAPPED TO DISCIPLINE								
		Systems Engineering (EV01)	Vehicle Management (EV40)	Structures & Environments (SIE) (EV30)	Propulsion (ER01)	Production (EM01)	Integrated Avionics and Software (ES01)	Operations (EO01)	Test (ET01)	S&MA (QD01)
		[EV70, EE12]	[EV40]	[EV30, ER40, ES21, ES22]	[ALL ER EXCEPT ER40]	[ALL EM]	[ALL ES EXCEPT ES21, ES22]	[ALL EO, ES10]	[ALL ET]	[ALL QD]
SLS Program Manager SLS Program Deputy Manager SLS Associate Program Manager Assistant PM Procurement	Program Chief Engineer Program Deputy Chief Engineer SE&I Technical Manager Assistant CE for Affordability Tech. Assist. Cross Program Integ. Tech. Assist. Ext. Interface Integ.	LSE: EV01 Alt: EV70 Alt: EV73	DLE: EV40 Alt: EV40	DLE: EV30 Alt: EV30	DLE: ER01 Alt: ER51 Alt: ER24	DLE: EM03 Alt: EM03	DLE: ES30 Alt: ES01	DLE: EO04 Alt: EO04	DLE: ET10 Alt: ET10	Program CSO Deputy CSO QD05 SE&I S&MA Lead QD05
Stages Element Manager Stages Deputy Element Manager Avionics Manager Core Stage Manager Integration Manager	Stages Chief Engineer Stages Deputy Chief Engineer Stages Deputy CE - Avionics Stages Deputy Chief Engineer - Test	EV70 Alt: EV71	EDLE: EV41	EDLE: EV34	EDLE: ER22	EDLE: EM03 Alt: EM32	EDLE: ES12	EDLE: EO40	EDLE: ET10	QD03
Booster Element Manager Booster Deputy Element Manager Control Systems Manager Assess & Trust Systems Manager Motor/BSM ASM Booster CE/Interface Mgr	Booster Chief Engineer Booster Deputy Chief Engineer	ER50	EDLE: EV40	EDLE: ER40	EDLE: ER51	EDLE: EM03	EDLE: ES12	EDLE: EO40		QD01
Engines Element Manager Engines Deputy Element Manager	Engines Chief Engineer Engines Deputy Chief Engineer	ER20	EDLE: EV43	EDLE: ER41	EDLE: ER21	EDLE: EM03	EDLE: ES12	EDLE: ER21		QD02
Spacecraft/Payload Integration and Evolution (SPIE) Office Manager SPIE Deputy Manager	SPIE CE SPIE Deputy CE	EV70 Alt: EV70	EDLE: EV41	EDLE: EV30	EDLE: ER23	EDLE: EM03	EDLE: ES10	EDLE: EO40	EDLE: ET30	QD02
	SPIE CE SPIE Deputy CE				EDLE: ER01 Alt: ER21	EDLE: EM03				QD01

◆ Margin is maintained by the Organization, not in the margin management tables

- Biased Information Sharing
- Margin Management is focused on Managing the Disciplines (informed by the System Integrating Physics)

◆ SLS Organizational Structure was defined by the LSE as a recommendation to the Chief Engineer and the Program Manager

Discipline Integration Models



Organizational Values

Goal Function Tree (GFT)

Value Model

Goals

Value Attributes

System Functions

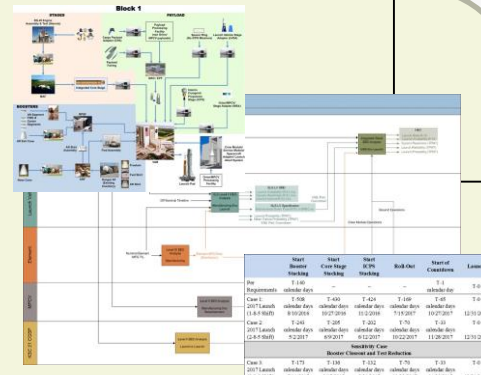
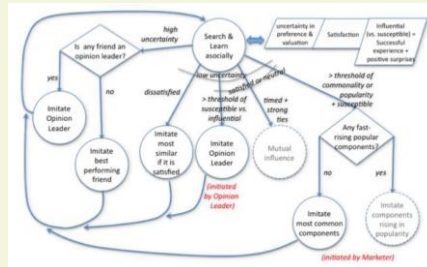
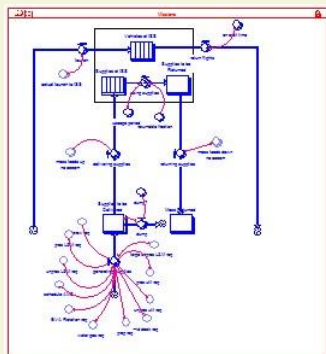
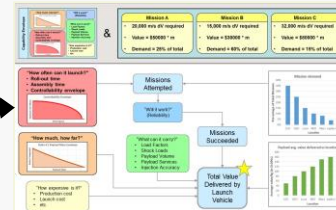
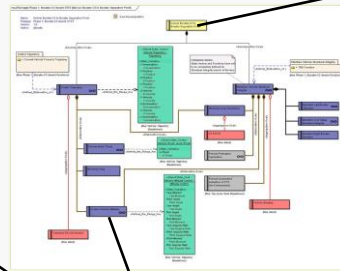
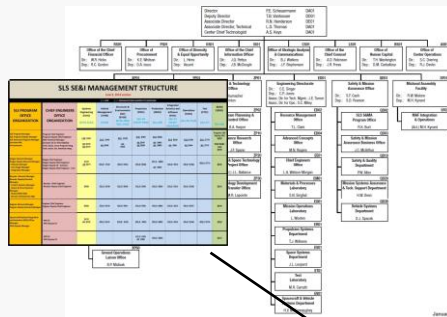
- Allow systems engineers to:
 - Understand information flow through the development and/or operations organization
 - Integrate discipline information into a system level design
 - Analyze information flow, gaps, and blind spots at the System level

Agent Based Model (ABM)

Discrete Event Simulation

System Dynamics Model

- MagicDraw Enterprise (SysML)
- Matlab
- Matlab StateFlow
- JAVA
- Anylogic
- Extend



Summary



- ◆ **Discussed approach to Engineering an Elegant System**
- ◆ **Systems Engineering Framework and Principles**
 - System Integration
 - Engineering Discipline Integration
- ◆ **Several methods and tools are available for conducting integrated system design and analysis**
 - System Integration
 - System State Variables
 - Goal Function Tree
 - State Analysis Model
 - System Value Model
 - System Integrating Physics
 - Topics Not Discussed
 - System Autonomy
 - Multidisciplinary Design Optimization (MDO)
 - Engineering Statistics
 - Discipline Integration
 - Sociological Concepts in Systems Engineering
 - Information Flow
 - Topics Not Discussed
 - Systems Thinking (Cognitive Science)
 - Policy and Law
 - System Dynamics Modeling
- ◆ **Systems Engineering Approach defined in two documents**
 - “Engineering Elegant Systems: Theory of Systems Engineering”
 - “Engineering Elegant Systems: The Practice of Systems Engineering”
- Send requests for documents to: michael.d.Watson@nasa.gov

Backup

